Final

Due: December 18, 2024

CSCI 1510: Intro. to Cryptography and Computer Security

- The final exam is due at 11:59 PM on December 18th (Wednesday). **No late days or extensions will be granted.**
- You may consult the course materials and textbooks, but you must write each answer in your own words/structure. Apart from that, you may *not* collaborate or discuss problems with the instructor or TAs.
- If you have any clarifying questions on the exam, please post a private post on EdStem, and we will respond as soon as we can (within a day).

1 Warm-Ups (10 points)

a. (1 point) DDH assumption [ ] (does/does not) imply DLOG assumption.

b. (1 point) CDH assumption [ ] (does/does not) imply DDH assumption.

c. (1 point) RSA assumption [ ] (does/does not) imply the factoring assumption.

d. (1 point) For public-key encryption schemes, semantic security [ ] (does/does not) imply CPA security.

e. (2 points) Suppose an eavesdropper adversary $\mathcal{A}$ observes two ElGamal encryptions (under the same key): $c_a = \langle c_{a,1}, c_{a,2} \rangle$ and $c_b = \langle c_{b,1}, c_{b,2} \rangle$, where c_a is an encryption of (unknown) m_a and c_b is an encryption of (unknown) m_b. Without knowing the secret key or the two messages m_a, m_b, $\mathcal{A}$ can construct a new ciphertext $c = \langle c_1, c_2 \rangle$ that is an encryption of a new messages $m = (m_a \cdot m_b)^2$. How can $\mathcal{A}$ construct c from c_a and c_b? In particular, define c_1 and c_2 that compose c.

$c_1 =$ [ ] ;
$c_2 =$ [ ] .

f. (2 points) What are the two steps in constructing a fully homomorphic encryption (FHE) scheme?

Step 1: [ ];
Step 2: [ ].

g. (2 points) What are the 4 properties that a group $(\mathbb{G}, \cdot)$ must satisfy? Include mathematical descriptions.

2 Decisional Diffie-Hellman Assumption and PRGs (11 points)

Let $\mathbb{G}$ be a cyclic group of prime order q with generator g.

a. (8 points) Let $w = p(n)$ where n is the security parameter and $p(\cdot)$ is some polynomial. Prove that under the DDH assumption, the following two distributions are computationally indistinguishable:

$$\left(g^{a_1}, g^{a_2}, \ldots, g^{a_w}, g^{a_1 \cdot b}, g^{a_2 \cdot b}, \ldots, g^{a_w \cdot b}\right) \overset{c}{\approx} \left(g^{a_1}, g^{a_2}, \ldots, g^{a_w}, g^{c_1}, g^{c_2}, \ldots, g^{c_w}\right),$$

where $a_i, b, c_i \overset{\$}{\leftarrow} \mathbb{Z}_q$ are sampled independently and uniformly at random.

b. (3 points) Using what you proved in part (a), construct a PRG $G : \mathbb{Z}_q^{w+1} \to \mathbb{G}^{2w}$ with w defined in part (a). You do not need to provide a formal proof, but briefly explain how your answer from (a) connects to your construction.

3 Composing One-Way Functions (15 points)

Let $f : \{0,1\}^n \mapsto \{0,1\}^n$ be a one-way function. Define $f^{(2)} : \{0,1\}^n \mapsto \{0,1\}^n$ as

$$f^{(2)}(x) \coloneqq f(f(x)).$$

We say that we can securely compose f if $f^{(2)}$ is also one-way.

a. (5 points) Prove that if f is a one-way permutation, then $f^{(2)}$ is also a one-way permutation.

b. (5 points) In this part of the problem, we will construct a one-way function from a pseudorandom generator.

   For simplicity, assume n is even. Let $G : \{0,1\}^{n/2} \mapsto \{0,1\}^n$ be a pseudorandom generator. Define $f : \{0,1\}^n \mapsto \{0,1\}^n$ as $f(x) \coloneqq G(x_1)$, where x_1 is the first $n/2$ bits of x.

   Prove that, if G is a PRG, then f is a one-way function.

c. (5 points) Prove that f constructed in part (b) can be securely composed, namely $f^{(2)}$ is also a one-way function.

4 Signatures from Bilinear Pairings (15 points)

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two cyclic groups, both of prime order q, with generators $g \in \mathbb{G}$ and $g_T \in \mathbb{G}_T$, respectively. A Type-I *bilinear pairing* is an efficiently computable function $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ satisfying the following properties:

- $e(g, g) = g_T$;
- For any $a, b \in \mathbb{Z}_q$, $e(g^a, g^b) = {g_T}^{a \cdot b}$.

This makes the DDH problem in $\mathbb{G}$ easy. To see why, given a DH tuple (g^a, g^b, g^c), one can test if $c = a \cdot b$ by checking if $e(g^a, g^b) \stackrel{?}{=} e(g^c, g)$. Nevertheless, we can still assume the CDH problem is hard in $\mathbb{G}$. That is, given (g^a, g^b) for $a, b \xleftarrow{\$} \mathbb{Z}_q$, it is computationally hard for any PPT algorithm to find g^{ab}.

We construct a signature scheme using the bilinear pairing described above and a hash function $H : \{0,1\}^* \to \mathbb{G}$, modeled as a random oracle.

The initialization process first generates all the bilinear pairing parameters $(\mathbb{G}, \mathbb{G}_T, q, g, g_T, e)$. The signature scheme is constructed as follows.

- $\mathsf{Gen}(1^n)$: Sample $x \xleftarrow{\$} \mathbb{Z}_q$. Output $\mathsf{sk} = x$ and $\mathsf{pk} = g^x$.
- $\mathsf{Sign}_{\mathsf{sk}}(m)$ for $m \in \{0,1\}^*$: Compute $h = H(m)$, and output $\sigma = h^{\mathsf{sk}}$.
- $\mathsf{Vrfy}_{\mathsf{pk}}(m, \sigma)$: Compute $h = H(m)$, and verify if $e(h, \mathsf{pk}) = e(\sigma, g)$.

a. (3 points) Prove the correctness of this signature scheme, i.e., an honestly computed signature on a message will always verify.

b. (12 points) Assuming the CDH problem is hard in $\mathbb{G}$, prove that this signature scheme is secure in the random oracle model.

5 Zero-Knowledge Proof for 3SAT (15 points)

Recall the NP-complete language 3SAT. A Boolean formula over variables $x_1, \ldots, x_n$ is 3-CNF if we can write $\phi = \phi_1 \wedge \phi_2 \wedge \ldots \wedge \phi_m$ where each $\phi_i = y_{i,1} \vee y_{i,2} \vee y_{i,3}$, and each $y_{i,j}$ is a literal over $x_1, \ldots, x_n$, i.e., it is in the set of formulas $\{x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n\}$. For example, $(x_1 \vee x_2 \vee \bar{x}_3) \wedge (\bar{x}_2 \vee x_4 \vee \bar{x}_5)$ is a 3-CNF formula over $(x_1, x_2, x_3, x_4, x_5)$.

A Boolean formula ϕ is satisfiable if there exists an assignment $(a_1, \ldots, a_n)$ to its variables such that $\phi(a_1, \ldots, a_n) = 1$. More formally, consider the language

$$\text{3SAT} = \{ \text{ 3-CNF formula } \phi \mid \exists a_1, \ldots, a_n \text{ such that } \phi(a_1, \ldots, a_n) = 1 \}$$

Construct a zero-knowledge proof system for 3SAT. Prove that the proof system you give is complete and sound. Additionally, start the proof for the zero-knowledge property by providing the construction of a simulator (you do *not* need to finish the proof).

You may use commitment schemes, but may *not* reduce this problem to any other NP-complete problem.

6 Oblivious Transfer (8 points)

Let $\mathcal{F} = \{f_i : D_i \to R_i\}_{i \in \mathcal{I}}$ be a trapdoor permutation with a hard-core predicate hc. Let Gen be the sampling algorithm and Inv be the invert algorithm for $\mathcal{F}$. Consider the following oblivious transfer protocol.

Sender A's Input: (m_0, m_1) where $m_0, m_1 \in \{0,1\}$. Both messages are single bits.

Receiver B's Input: $c \in \{0,1\}$.

- A runs $(i,t) \leftarrow \mathsf{Gen}(1^n)$ and sends i to B.
- B samples $x \xleftarrow{\$} D_i$ and computes $y_c := f_i(x)$. B also samples $y_{1-c} \xleftarrow{\$} R_i$, and sends (y_0, y_1) to A.
- A computes $z_j := \mathsf{hc}_i(\mathsf{Inv}(i,t,y_j)) \oplus m_j$ for $j \in \{0,1\}$, and sends (z_0, z_1) to B.
- B outputs $m_c := \mathsf{hc}_i(x) \oplus z_c$.

Start the proof of *semi-honest* security by providing constructions of the simulator for both parties. You do *not* need to finish the proof.

a. (4 points) Construct a simulator that simulates the view for the sender A.

b. (4 points) Construct a simulator that simulates the view for the receiver B.

7 Discussion (6 points)

a. (3 points) Describe an arbitrary potential application of secure two-party or multi-party computation.

b. (3 points) Describe an arbitrary potential application of somewhat or fully homomorphic encryption.